STEP 1: login to the Linux system on <u>Linuxlab server</u>. Start a terminal (the shell prompt).



Fig. 1 The screen when you login to the Linuxlab through equeue

STEP 2: In the terminal, execute the following command:

## module add ese461

You could perform "*module avail*" in the terminal to find the available modules on Linuxlab. Make sure ese461 is presented when you execute this command.

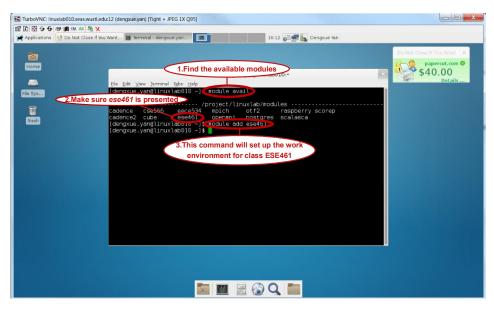


Fig. 2 Build work environment for class ESE461 using module

STEP 3: Getting started with Verilog

- Creating a new folder (better if you have all the files for a project in a specific folder).
- Enter into this new folder and start writing your Verilog script in a new file (.v file). Example code for modeling an counter is <u>here</u>
- In addition to model code, Test Bench script has to be given in order to verify the functionality of your model (.v file). Example code of test bench for counter is <u>here</u>.

Terminal – dengxue.yan@linuxlab010:*/ESE461/VcsTutorial Eile Edit ⊻iew Ierminal T <u>a</u> bs Help	×
[dengxue.yan@linuxlab010 ~]\$ module add ese461	
[dengxue.yan@linuxlab010 ~]\$ cd ESE461/VcsTutorial/	
[dengxue.yan@linuxlab010 VcsTutorial]; gedit Counter.v &	Use gedit to edit the .v files
[dengxue.yan@linuxlab010 VcsTutorial]; gedit Counter_tb.v &	gedit is a commonly used GUI editor on Linux
[2] 24266	
[dengxue.yan@linuxlab010 VcsTutorial]\$ 🚪	

Fig. 3 Open gedit through teminal

STEP 4: Compiling and simulating your code

• In the terminal, change the directory to where your model and test bench files (Counter.v and Counter\_tb.v) are present by using this command:

cd <path> For example: cd ~/ESE461/VcsTutorial/ (Remark: '~' means home directory on Linux)

• Compile the files by typing in the terminal:

vcs <file>.v <file\_tb>.v

In the above example, it should be:

vcs Counter.v Counter\_tb.v

There should be no error presented in the terminal. Otherwise you need to check your code and correct them according to the related message. The complier will print out detailed information about your mistakes in the code.



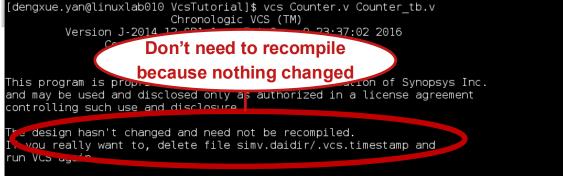


Fig. 5 The result of recompile the code when nothing changed

[dengxue.vape] :	
Error message when encounter a <sup>23:39:13</sup> <sup>2016</sup>	1
error during the compiling process.	
This prog. and may be used and disclosed only as authorized in a license agreement controlling such use and disclosure.	
Parsing design file 'Counter.v'	
Frror-[PNDIID] Port not defined in IO declaration Counter.v, 3 Identifier 'rst' is not defined in IO declaration Source info: : rst Finase refer to LRM [1364-2001], section 12.3.3.	
Parsing design file Country_conv	
CPU time: .105 seconds to compile [dengxue.yan@linuxlab010 VcsTutorial]\$	

Fig.6 The result of vcs when it thinks there are mistakes in the code

- A successfully compiling will print out on terminal "../simv up to date". And it should generate an executable file named "simv" in the same folder where your codes are present.
- Then in the terminal run: ./simv
- After the process finishes, "VCS Simulation Report" will be present on the terminal and a file named "<file>.vcd" will be generated in the same folder where your codes are present. This is the dump file we specified in the test bench code and we will use it to graphically display the simulation results.

[dengxue.yan@linuxlab010 VcsTutorial]\$ ./simv
Chronologic VCS simulator
Contains Synopsys p Simulation Report
Compiler Version J-201
\$finish called from file "Counter ip.v", line 37.
\$finish at simulation time iscore
<pre><vcs pre="" report<="" simulation=""></vcs></pre>
Time: 1300000 ps
CPU Time: 0.360 seconds; Data structure size: 0.0Mb
Fri Sep 9 23:51:26 2016
[dengxue.yan@linuxlab010 VcsTutorial]\$ 📕

Fig. 7 Simulation Report

STEP 5: Displaying your Results graphically using dve

 After simulation report and "<file>.vcd" is generated, now type the following command in the terminal: dve

This is a viewer to plot and verify your results.

(Remark: an "&" can be placed behind the command, which means this command will run in background, so the terminal will be released)

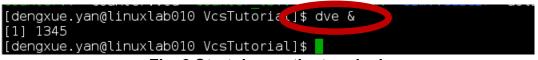


Fig. 8 Start dve on the terminal

• Go to "*File/Open Database*" and select the ".vcd" file from the project folder.

Turbs/WC Insolabilitions.wortlastu:12 (days	pair yani (Fight + IPEG 14 QM)	
※回日ののの書きます。		
Approahene 😲 De Net Cose if Veurifiert.	n Deller Hystered Le (Maar 🖉 Caurder Jour Le Millel L . 📓 Bernard - dergeset yer 📲 Bernard - dergeset yer 📑	
	048 - TopLevel.1 - (Source.1)	
and Dit you Gaulute Giguel Scope Date 20		ald a
Care Database. Canad	□ □ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
Com funct Contigen. Com funct Contigen. Com funct Contigen. Ender Contigen. Last gatem. Last gatem. Last gatem. Last gatem. Last gatem. Record Contigen. Contigency Contigency Contigency Contigency Contigency Contigency Last gatem. Last gatem		
		P Sector
2 Same [21 B Severity [21 B Code [44		2
demo	2	an an Mathemat

## Fig. 9 dve open database (1)

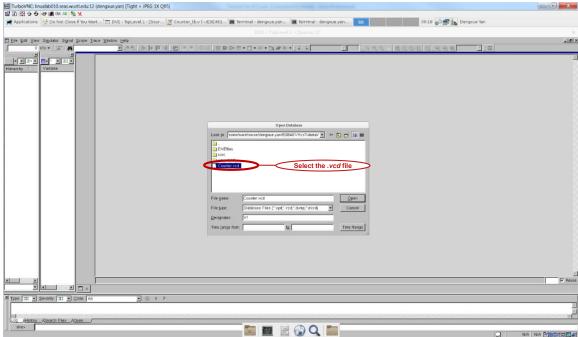


Fig. 10 dve open database (2)

• Then you will find the name of your test bench model in the *Hierarchy* box (*Counter\_tb* here). Expand it so that you can find *DUT* in the options.

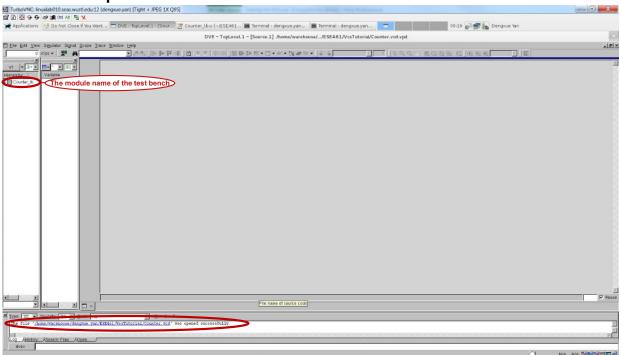


Fig. 11 dve open database (3)

• If you click on *DUT*, select the signals listed(all or partial) and right click, you will find an option "*Add to Waves*".

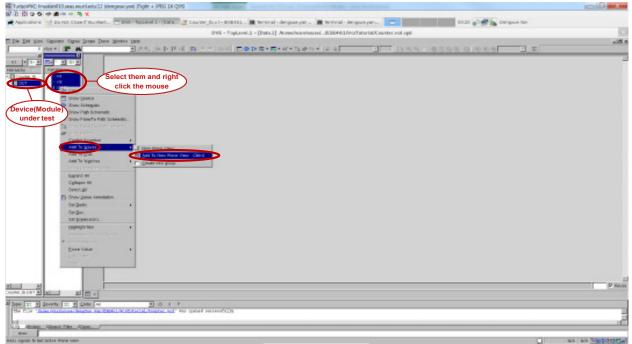


Fig. 12 dve open database (4)

• Click on "*Add to New Wave View*" to see the waveforms of your Inputs and Outputs. You should see your results in a new window. Then adjust the size of the waveform and explore other options as well.



Fig. 13 The waveform display

## Additional Option to run in Debug mode:

Instead of compiling the files directly as before, we can enable a debug flag during compilation by using following command

vcs -lca -debug\_access+all Counter.v Counter\_tb.v

Now run the code:

./simv -gui &

This should open the *dve* tool automatically and you can fully run your test bench or debug it step by step. To do this first select inputs and outputs from variable window and right click "*Add to the Waves*" as before. This should open the following window as shown in the Fig. 14 . Then click the tool button of blue arrow in brace or press *F11* to run the test bench step by step(Fig. 14 and Fig. 15). Or click the tool button of the blue arrow pointing downward or press *F5* to run the test bench fully(Fig. 16 ~ Fig. 18). Other tool options are also available and just explore them by yourself.



🚰 TurboVNC: Illinuxiabluo. Seas. Wusika 22:08 💦 🚮 DVE - TopLevel 2 - [Wave.1] /h / /ESE461/ - 19 **1** 19 19 1 • 沈 🖉 🕸 🔹 👗 🗛 Edg P P -6 (1) - 2- -• == • 👯 nere or press "F11" to ru test bench step by ster + oc#0 13 14 15 16 17 18 19 20 21 22 23 24 25 [4:0] c; [4:0] c = 5:500 The cursor indicates which line of code the ulator is executing when click "Step " but Wave.1 × B) to stopped Counter\_to.DUT 470000ps

Fig. 14 The waveform display in debug mode

Fig.15 The code trace cursor in debug mode

TurboVNC: linuxlab006.seas.wustl.edu:13 (dengxue.yan) [Tight + JPEG 1X OS	5]	_ 0 %
1 🗈 🔁 🕂 🔂 🕼 🏨 cm At 🐁 🗙		
🛃 Applications 🛛 🧐 Do Not Close If You Want 🔄 DVE - TopLevel.1 - [Data	🗖 DVE - TopLevel 2 - [Waxe 🗮 Terminal - dengsue yan 📃 19:45 🚛 🚝 🛼 Dengsue Yan	
	DVE - TopLeveL2 - [Wave.1] /home/warehouse//ESE461/VcsTutorial/simv	×
Eile Edit View Sigulator Signal Scope Trace Window Help	錢 Elle Edt Niew Sitünistor Sidual Sicobe Disce Mundaw Helb	<u>8</u> ×
1300000 X1px A 4		1
		C1:1300000
Sin + 12 S todule Counter_th.		REF
Hierarchy Variable 14 reg sat;	A Section 1. Click here or press provide provi	20000
Diff (200 + cR 17 wire [4:0] c:	"F5" to run simulation	
*-*-C[40] 18 19 Dounter DUT(		Innnnnn
20 .rot(rot), 21 .elk(clk),		0000000000
22	New Group	
84 25 initial		
25 initial 26 begin 27		19
28 Sduapfile ("Counter. vod" 29 Sduapwara (0, Counter. th		
30 31 xst = 1;		
32 clk = 0; 33		
34 #40		
Goto contraction of the second s	A A A A A A A A A A A A A A A A A A A	
Type: III - Severity III - Code: All		
Contains Despays proprietary information. Contains Version 7-2014 12-5P1-1, Buntime version J-2014.12-5P1-1, 5	n 1	
(a) Viter J-2014 12-SP1-1 Copyright (c) 1991-2014 by Spropsys Inc. The file '/none/warehouse/decayne.yun/ESE461/VcrDutorial/inter.ypd' w		
Sfinish called from file "Dounter th v", line 46. Sfinish at simulation time 1300000		
Simulation couplets, the is 1300000 pp. VCS Simulation Report		
The 1300000 ps GPU Time 0.100 seconds, Data structure size: 0.100		
Sun Sep 11 18:40.00 seconds; Data structure size: 0.000		
2. The report indicates the simulation has finishe		
		10
	Televis and anothe second formation second theory and the second televis	1200000
	00001( 000004) CO0004) SCC007( CO0034) SCC007 800302( 800302) 800302( 800301) 4	
V.og Atlistory ASearch Files ASearch Identifiers AOpen.	N Wave 1	
dves	UI p frished Counter to 13000	
	(II) • finished Counter to 13000	CODE REALESCONTING

Fig.16 Fully "run" in debug mode(1)

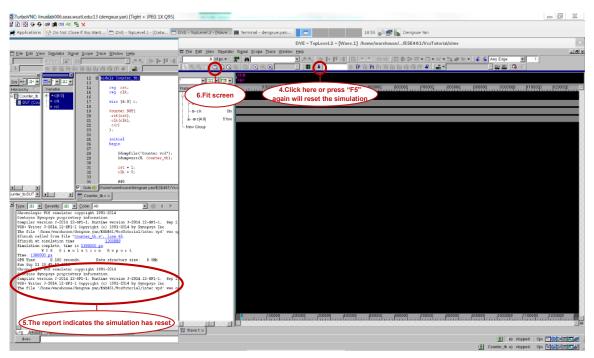


Fig.17 Fully "run" in debug mode(2) – simulation reset

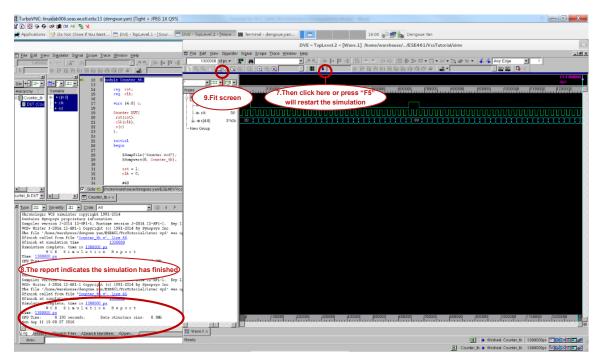


Fig.18 Fully run in debug mode(3) -- simulation restart

## **Reference:**

If you are interested in exploring further, another example model and test bench codes are present in the following link

https://github.com/bangonkali/electronics/tree/master/verilog/adder

Reference for test bench syntax can be found <u>here</u>.